Tutorons

Generating Context-Relevant, On-Demand Explanations and Demonstrations of Online Code

Andrew Head, Codanda Appachu, Marti A. Hearst, Björn Hartmann
Computer Science Division, UC Berkeley

unix command to fetch all tarball files from website

Google Search I'm Feeling Lucky

Automatically download music

This last technique, suggested by Jeff Veen, is by far my favorite use of Wget. These days there are tons of directories, aggregators, filters and weblogs that point off to interesting types of media. Using Wget, you can create a text file list of your favorite sites that say, link to MP3 files, and schedule it to automatically download any newly-added MP3's from those sites each day or week.

First, create a text file called mp3_sites.txt, and list URLs of your favorite sources of music online one per line (like http://del.icio.us/tag/system:fil... or stereogum.com). Be sure to check out my previous feature on how to find free music on the web for more ideas.

Then use the following Wget command to go out and fetch those MP3's:

```
wget -r -l1 -H -t1 -nd -N -np -A.mp3 -erobots=off -i mp3_sites.txt
```

That Wget recipe recursively downloads only MP3 files linked from the sites listed in mp3_sites.txt that are newer than any you've already downloaded. There are a few other specifications in there - like to not create a new directory for every music file, to ignore robots.txt and to not crawl up to the parent directory of a link. Jeff breaks it all down in his original post.

The great thing about this technique is that once this command is scheduled, you get an ever-rotating jukebox of new music Wget fetches for you while you sleep. With a good set of trusted sources, you'll never have to go looking for new music again - Wget will do all the work for you.

Install Wget

Wanna give all this a try? Windows users, you can download Wget here; Mac users, go here. An alternative for Windows users interested in more Linuxy goodness is to download and install the Unix emulator Cygwin which includes Wget and a whole slew of other 'nixy utilities, too.

Then use the following Wget command to go out and fetch those MP3's:

```
wget -r -l1 -H -t1 -nd -N -np -A.mp3 -erobots=off -i mp3_sites.txt
```

GNU Wget 1.16.2 Manual

Table of Contents

89 printed pages of the wget 'man' pages (12pt font)

Then use the following Wget command to go out and fetch those MP3's:

```
wget -r -l1 -H -t1 -nd -N -np -A.mp3 -erobots=off -i mp3_sites.txt
```

Then use the following Wget command to go out and fetch those MP3's:

```
wget -r -l1 -H -t1 -nd -N -np -A.mp3 -erobots=off -i mp3_sites.txt
```

Then use the following Wget command to go out and fetch those MP3's:

```
wget -r -l1 -H -t1 -nd -N -np -A.mp3 -erobots=off -i mp3_sites.txt
```

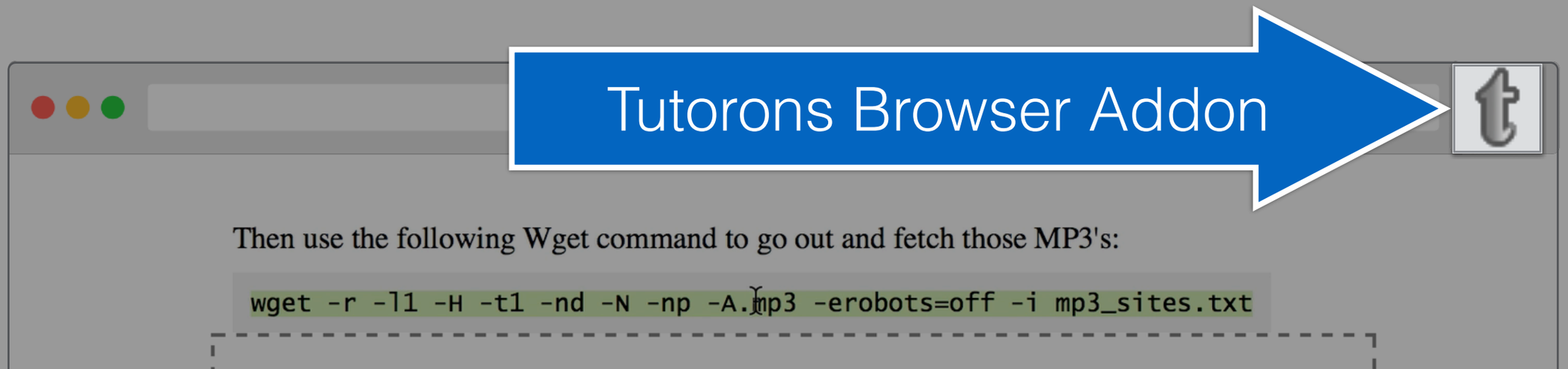
You found a *wget* command.

wget is a Terminal command you run to download a page from the Internet. Here, it downloads content from URLs from the file 'mp3_sites.txt'.

Recursively scrape web pages of type '.mp3' from URLs from the file 'mp3_sites.txt', following links 1 time.

It uses these options:

- `--recursive` (`-r`): specify recursive download.
- `--level` (`-l`): 1 is a maximum recursion depth (inf or 0 for infinite).
- `--span-hosts` (`-H`): go to foreign hosts when recursive.
- `--tries` (`-t`): set number of retries to 1 (0 unlimits).
- `--no-directories` (`-nd`): don't create directories.
- `--timestamping` (`-N`): don't re-retrieve files unless newer than local.
- `--no-parent` (`-np`): don't ascend to the parent directory.
- `--accept` (`-A`): .mp3 is a comma-separated list of accepted extensions.
- `--execute` (`-e`): execute a `.wgetrc'-style command (COMMAND=robots=off).
- `--input-file` (`-i`): download URLs found in local or external mp3_sites.txt.



Then use the following Wget command to go out and fetch those MP3's:

```
wget -r -l1 -H -t1 -nd -N -np -A.mp3 -erobots=off -i mp3_sites.txt
```

You found a *wget* command.

wget is a Terminal command you run to download a page from the Internet. Here, it downloads content from URLs from the file 'mp3_sites.txt'.

Recursively scrape web pages of type '.mp3' from URLs from the file 'mp3_sites.txt', following links 1 time.

It uses these options:

- `--recursive` (`-r`): specify recursive download.
- `--level` (`-l`): 1 is a maximum recursion depth (inf or 0 for infinite).
- `--span-hosts` (`-H`): go to foreign hosts when recursive.
- `--tries` (`-t`): set number of retries to 1 (0 unlimits).
- `--no-directories` (`-nd`): don't create directories.
- `--timestamping` (`-N`): don't re-retrieve files unless newer than local.
- `--no-parent` (`-np`): don't ascend to the parent directory.
- `--accept` (`-A`): .mp3 is a comma-separated list of accepted extensions.
- `--execute` (`-e`): execute a `.wgetrc'-style command (COMMAND=robots=off).
- `--input-file` (`-i`): download URLs found in local or external mp3_sites.txt.

An Explanation Built by a Tutoron

You found a *wget* command.

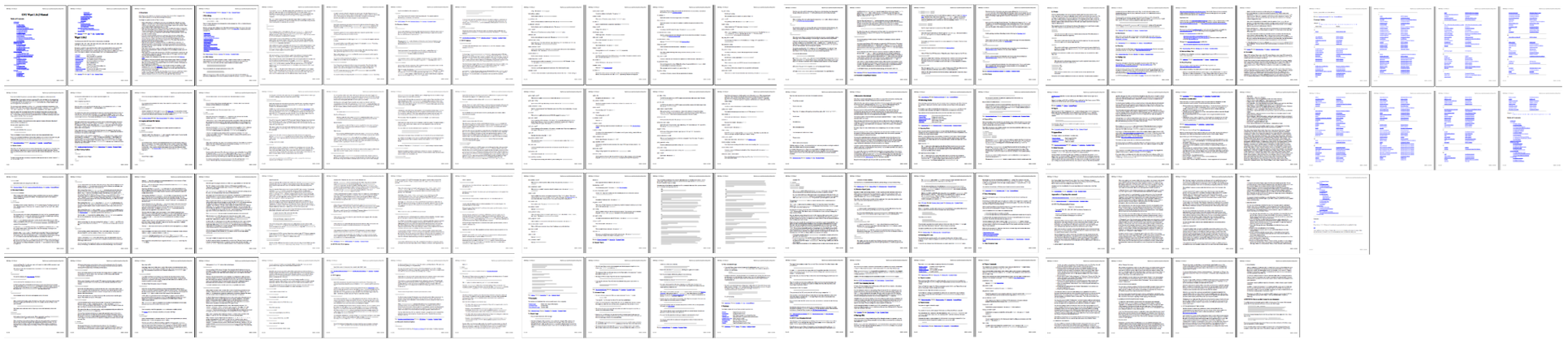
wget is a Terminal command you run to download a page from the Internet. Here, it downloads content from URLs from the file 'mp3_sites.txt'.

Recursively scrape web pages of type '.mp3' from URLs from the file 'mp3_sites.txt', following links 1 time.

It uses these options:

- `--recursive` (`-r`): specify recursive download.
- `--level` (`-l`): 1 is a maximum recursion depth (inf or 0 for infinite).
- `--span-hosts` (`-H`): go to foreign hosts when recursive.
- `--tries` (`-t`): set number of retries to 1 (0 unlimits).
- `--no-directories` (`-nd`): don't create directories.
- `--timestamping` (`-N`): don't re-retrieve files unless newer than local.
- `--no-parent` (`-np`): don't ascend to the parent directory.
- `--accept` (`-A`): .mp3 is a comma-separated list of accepted extensions.
- `--execute` (`-e`): execute a `.wgetrc'-style command (COMMAND=robots=off).
- `--input-file` (`-i`): download URLs found in local or external mp3_sites.txt.

What Explanations Can a Tutoron Build?



VS.

Low Volume of Text to Sift Through

Recursively scrape web pages of type '.mp3' from URLs from the file 'mp3_sites.txt', following links 1 time.

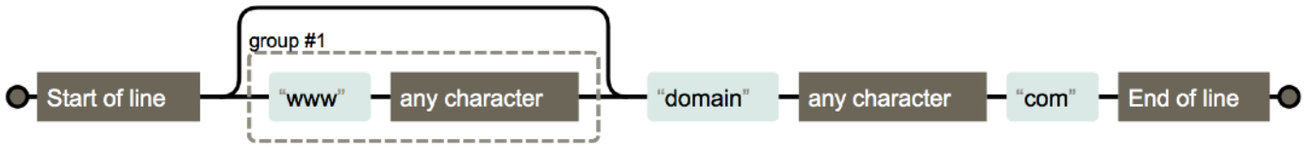
It uses these options:

- `--recursive` (`-r`): specify recursive download.
- `--level` (`-l`): 1 is a maximum recursion depth (inf or 0 for infinite).
- `--span-hosts` (`-H`): go to foreign hosts when recursive.

What Explanations Can a Tutoron Build?

VS.

Rich Explanations



Diagrams

```
<div class="content_container_7">
    <button>
    </button>
</div>
```

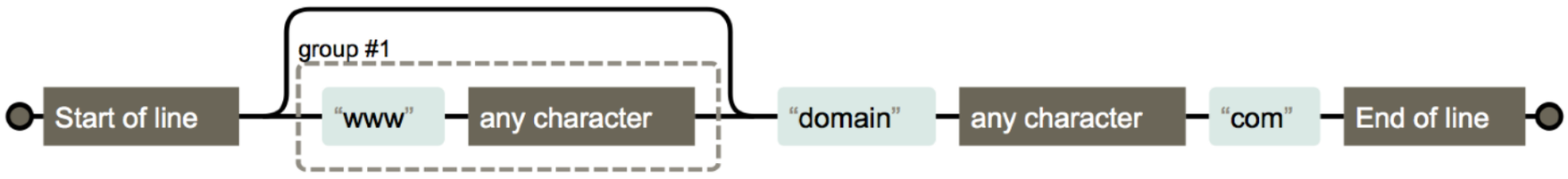
Usage Examples

> The selector '.content_container_7 button' chooses buttons from elements of class 'content_container_7'.

Synthesized Prose

What Explanations Can a Tutoron Build?

Diagrams



Usage Examples

```
<div class="content_container_7">
    <button>
    </button>
</div>
```

Synthesized Prose

The selector '.content_container_7 button' chooses buttons from elements of class 'content_container_7'.

Then use the following Wget command to go out and fetch those MP3's:

```
wget -r -l1 -H -t1 -nd -N -np -A.mp3 -erobots=off -i mp3_sites.txt
```

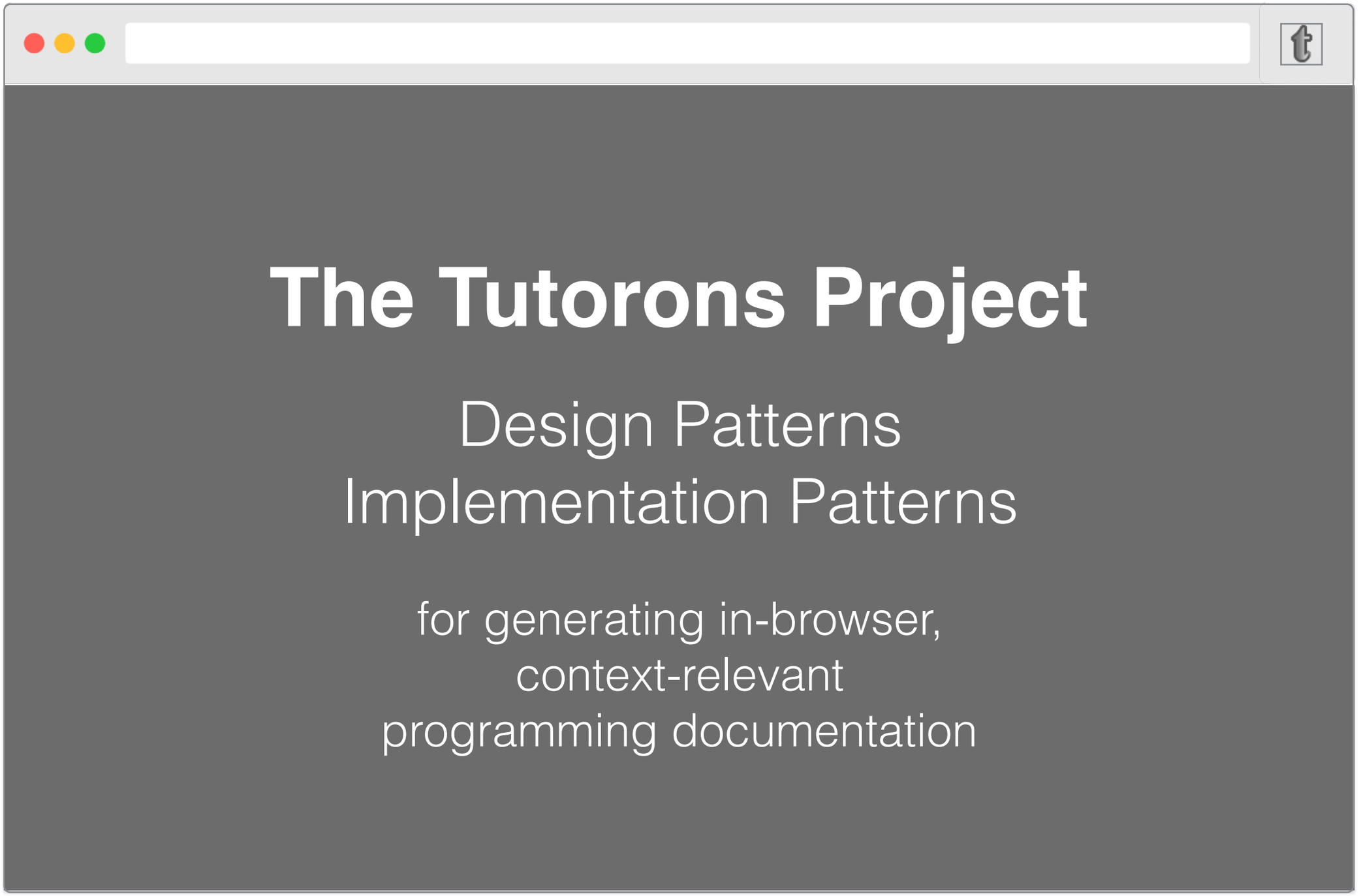
You found a *wget* command.

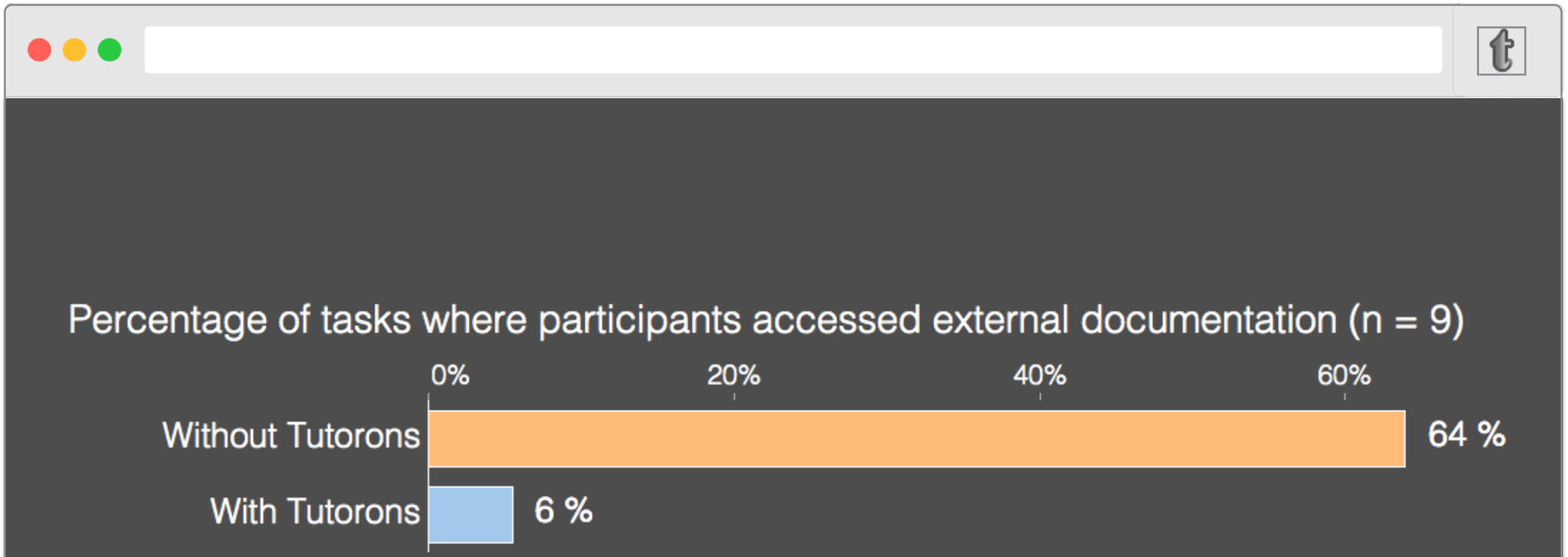
wget is a Terminal command you run to download a page from the Internet. Here, it downloads content from URLs from the file 'mp3_sites.txt'.

Recursively scrape web pages of type '.mp3' from URLs from the file 'mp3_sites.txt', following links 1 time.

It uses these options:

- `--recursive` (`-r`): specify recursive download.
- `--level` (`-l`): 1 is a maximum recursion depth (inf or 0 for infinite).
- `--span-hosts` (`-H`): go to foreign hosts when recursive.
- `--tries` (`-t`): set number of retries to 1 (0 unlimits).
- `--no-directories` (`-nd`): don't create directories.
- `--timestamping` (`-N`): don't re-retrieve files unless newer than local.
- `--no-parent` (`-np`): don't ascend to the parent directory.
- `--accept` (`-A`): .mp3 is a comma-separated list of accepted extensions.
- `--execute` (`-e`): execute a `.wgetrc'-style command (COMMAND=robots=off).
- `--input-file` (`-i`): download URLs found in local or external mp3_sites.txt.


The Tutorons Project
Design Patterns
Implementation Patterns
for generating in-browser,
context-relevant
programming documentation



Language	N	Precision	Recall
wget Unix command	203	95%	64%
CSS selectors	466	80%	41%
Regular expressions	445	70%	14%

This Talk

- Background
- Interacting with Tutorons
- Developing Tutorons

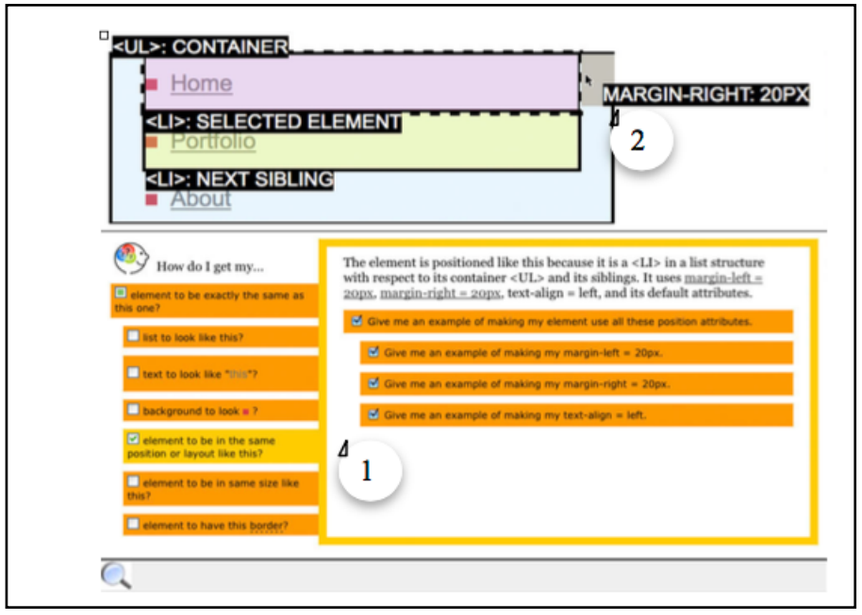
Over a two-hour programming session, participants spent an average **19%** of their programming time on the web.

(Brandt et al. 2009)

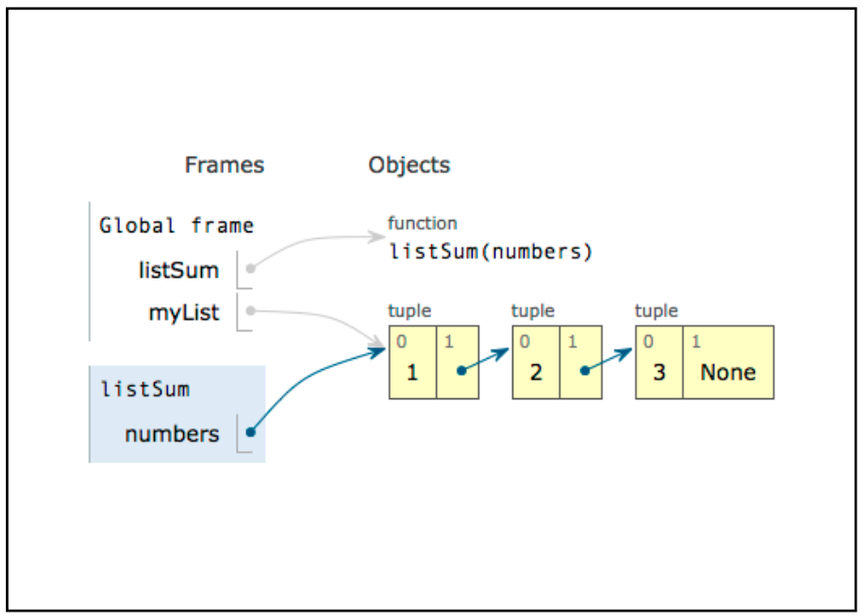
Programmers frequently ask questions like "What is the functionality of a given API type?"

(Duala-Ekoko & Robillard 2012)

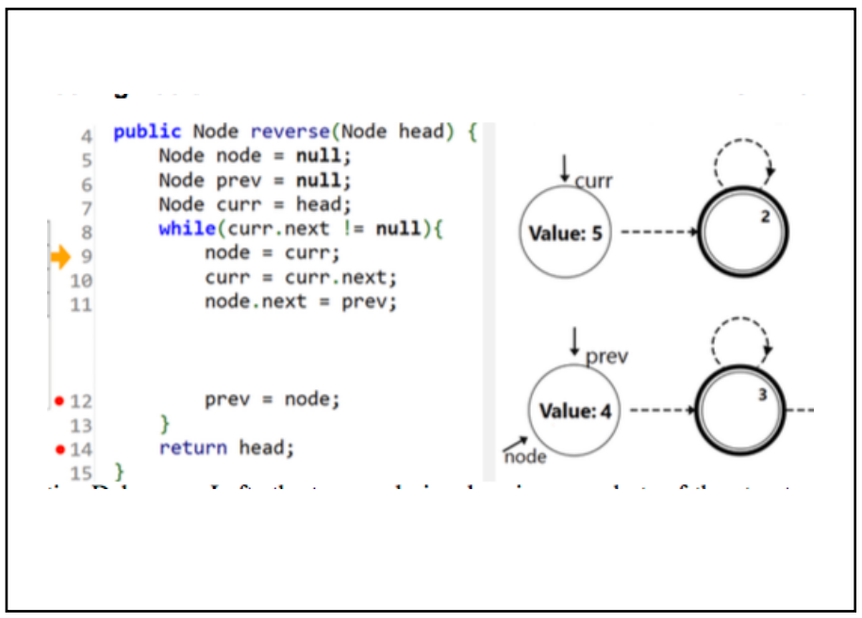
Demonstrating Code



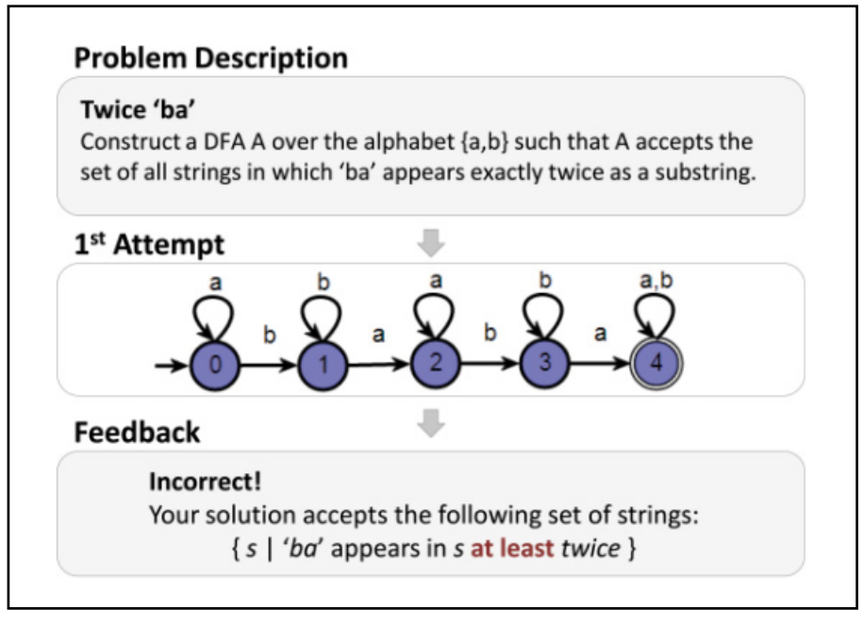
WebCrystal [Chang & Myers '12]



PythonTutor [Guo '13]

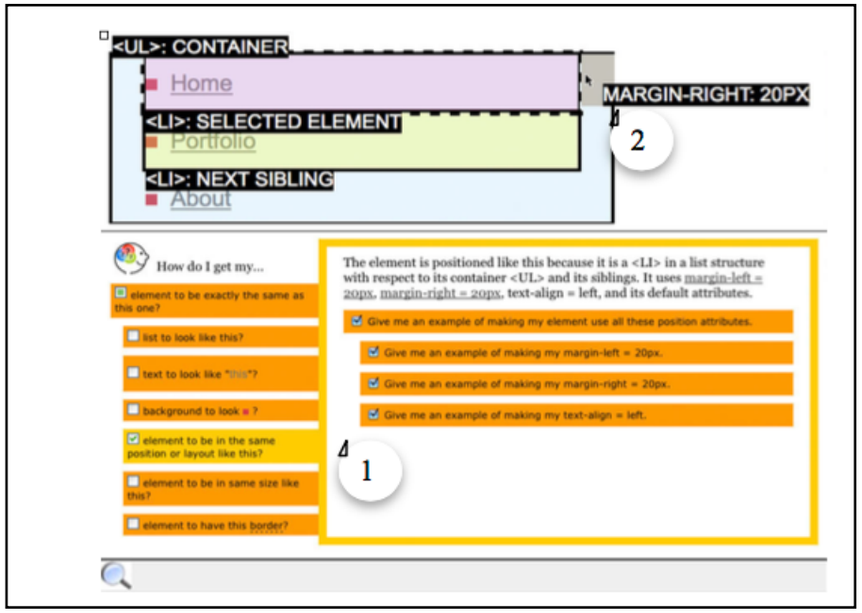


FluidEdt [Ou et al. '15]

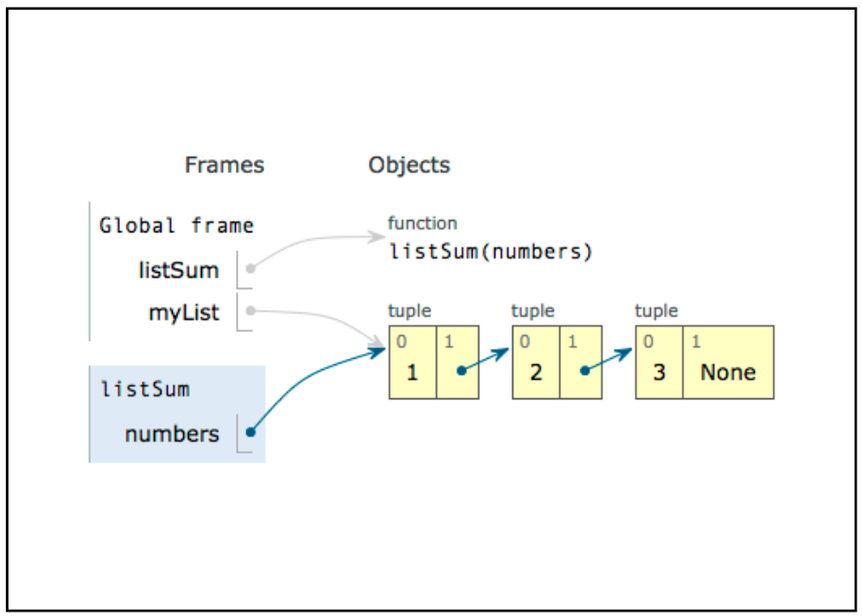


AutomataTutor [D'Antoni et al. '15]

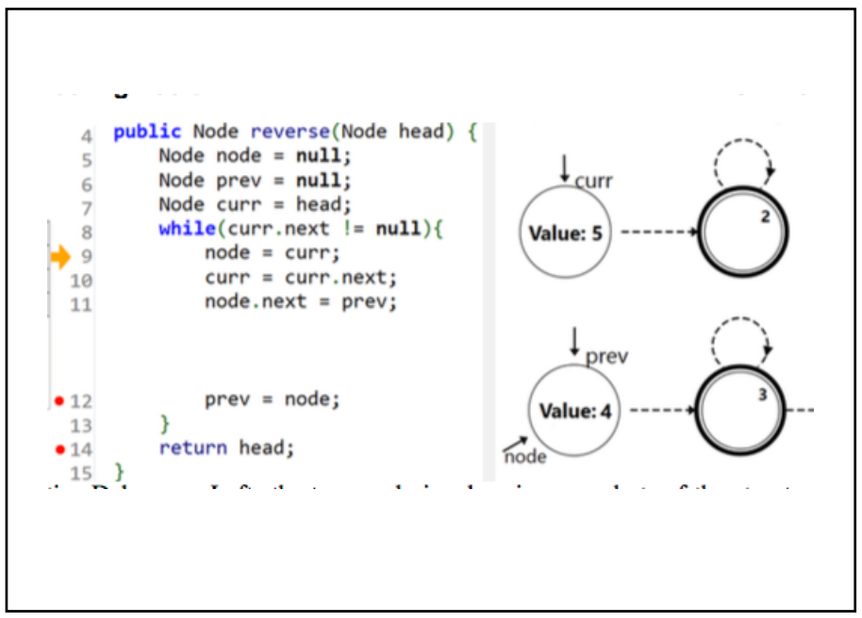
Demonstrating Code



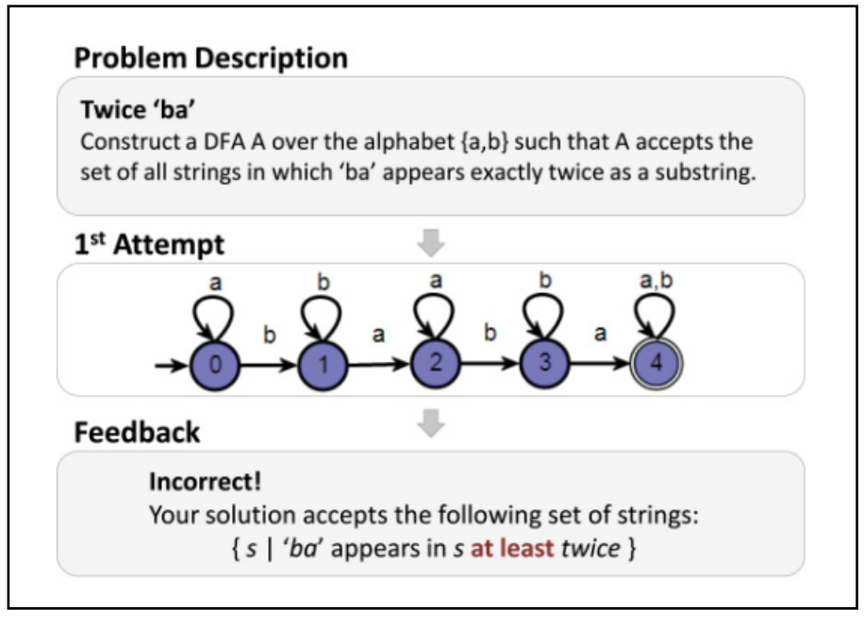
WebCrystal [Chang & Myers '12]



PythonTutor [Guo '13]



FluidEdt [Ou et al. '15]



AutomataTutor [D'Antoni et al. '15]

Explaining Code

```
fileName = showFileDialog(...);
```

"*show file dialog and get file name*"

Java methods (Sridhara et al. 2010)

```
onelineBox.add(f);
onelineBox.add(s);
onelineBox.add(t);
```

"*add given components to oneline box*"

Action sequences (Sridhara et al. 2011)

Explaining Code

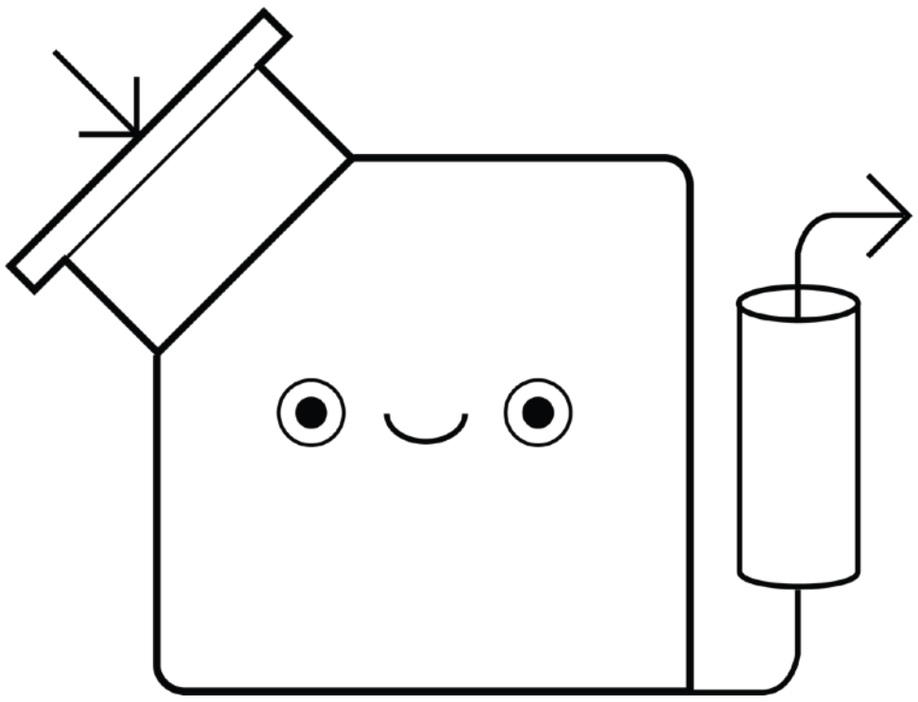
- Class diagrams (Burden & Heldal 2011)
- Parameters (Sridhara et al. 2011)
- Classes (Moreno et al. 2013)
- Unit test cases (Kamimura et al. 2014)
- Method context (McBurney & McMillan 2014)

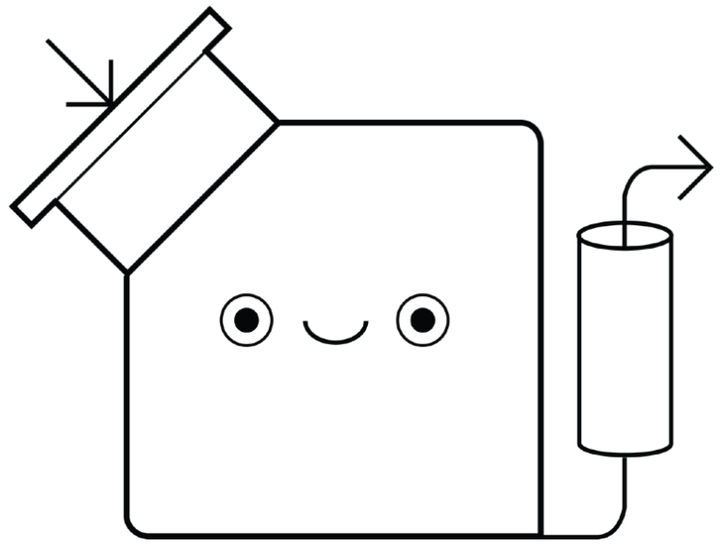
...

This Talk

- ~~Background~~
- **Interacting with Tutorons**
- Developing Tutorons

Tutoron. a routine on a web server with language-specific rules for detecting, parsing and explaining source code written on a web page.





Detect	Parse	Explain

Use m suffix for megabytes (--limit-rate=1m). The above command will limit the retrieval rate to 50KB/s. It is also possible to specify disk quota for automatic retrievals to avoid disk DoS attack. The following command will be aborted when the quota is (100MB+) exceeded.

```
$ wget -cb -o /tmp/download.log -i /tmp/download.txt --quota=100m
```

From the wget man page:

> Please note that Wget implements the limiting by sleeping the appropriate amount of time after a network read that took less time than specified by the rate. Eventually this strategy causes the TCP transfer to slow down to approximately the specified rate. However, it may take some time for this balance to be achieved, so don't be surprised if limiting the rate doesn't work well with very small files.

Use wget With the Password Protected Sites

You can supply the http username/password on server as follows:

```
$ wget --http-user=vivek --http-password=Secrete http://cyberciti.biz
/vivek/csits.tar.gz
```

Another way to specify username and password is in the URL itself.

```
$ wget 'http://username:password@cyberciti.biz/file.tar.gz
```

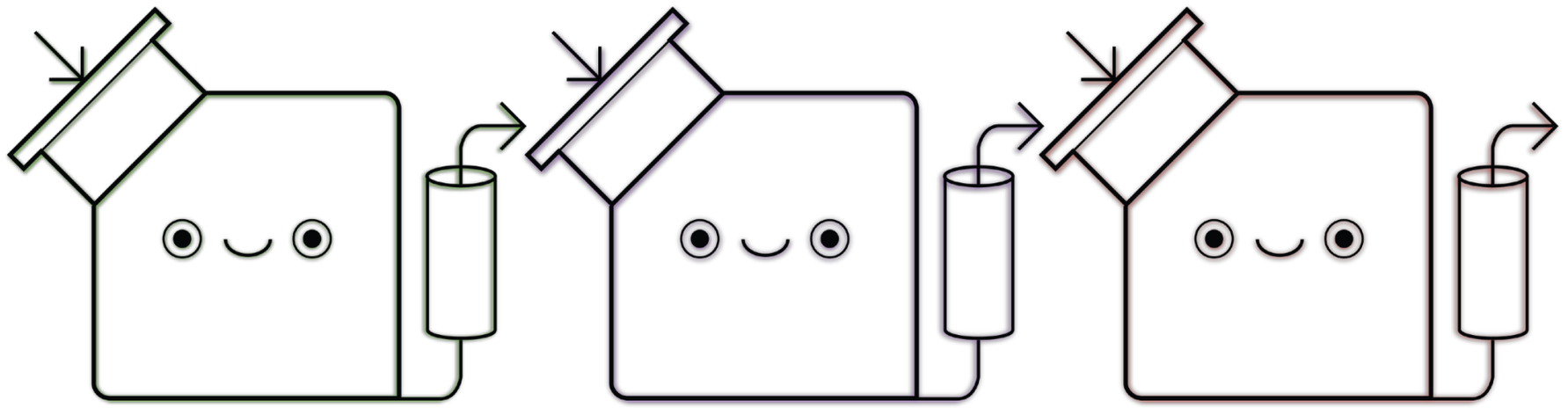
Option	Value
c	(null)
b	(null)
o	/tmp/download.log
i	/tmp/download.txt
quota	100m

You found a *wget* command.

wget is a Terminal command you run to download a page from the Internet. Here, it downloads content from URLs from the file '/tmp/download.txt'.

It uses these options:

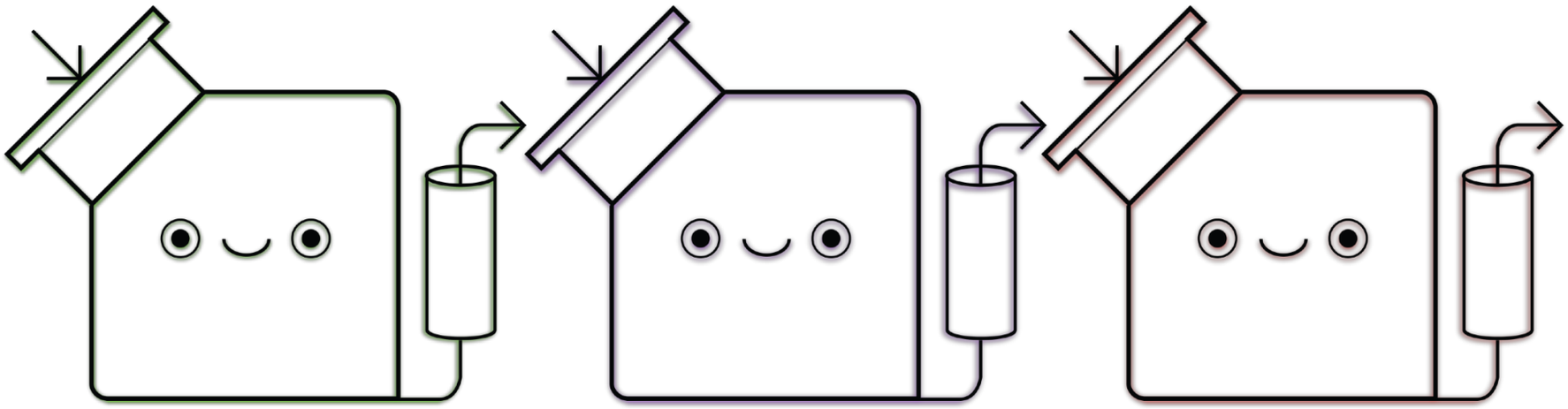
- `--continue` (`-c`): resume getting a partially-downloaded file.
- `--background` (`-b`): go to background after startup.
- `--output-file` (`-o`): log messages to /tmp/download.log.
- `--input-file` (`-i`): download URLs found in local or external /tmp/download.txt.
- `--quota` (`-Q`): set retrieval quota to 100m.



wget Unix command

CSS selectors

Regular expressions



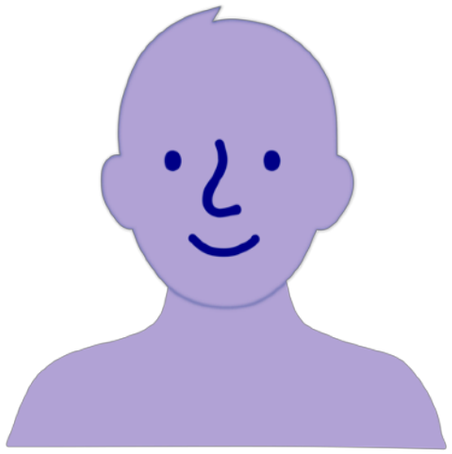
wget Unix command

CSS selectors

Regular expressions

And perhaps in the future:
LaTeX, Apache configs, SQL, etc.

Interacting with Tutorons



Programmer

Navigate

In-Situ Help


is one of my sessions!

From these pages we can scrape the session title, which appears at the top. We can also obtain the names of the speakers and the YouTube link from the sidebar that appears on the right side below the embedded video. The code that gets these elements is shown below:

```
def get_video_data(video_page_url):
    video_data = {}
    response = requests.get(root_url + video_page_url)
    soup = bs4.BeautifulSoup(response.text)
    video_data['title'] = soup.select('div#videobox h3')[0].get_text()
    video_data['speakers'] = [a.get_text() for a in soup.select('div#sidebar a[href^="/speaker/"]')]
    video_data['youtube_url'] = soup.select('div#sidebar a[href^="http://www.youtube.com"]')[0].get_text()
```

A few things to note about this function:

- The URLs returned from the scraping of the index page are relative, so the `root_url` needs to be prepended.
- The session title is obtained from the `<h3>` element inside the `<div>` with id `videobox`. Note that `[0]` is needed because the `select()` call returns a list, even if there is only one match.
- The speaker names and YouTube links are obtained in a similar way to the links in the index page.

Now all that remains is to scrape the views count from the YouTube page for each video. This is actually very simple to write as a continuation of the above function. In fact, it is so simple that while we are at it, we can also scrape the likes and dislikes counts:

```
def get_video_data(video_page_url):
    # ...
    response = requests.get(video_data['youtube_url'])
    soup = bs4.BeautifulSoup(response.text)
    video_data['views'] = int(re.sub('[^0-9]', '',
                                     soup.select('.watch-view-count')[0].get_text().split()[0]))
    video_data['likes'] = int(re.sub('[^0-9]', '',
```

Web Browser

Tutorons Library or Addon

I need files to be downloaded to /tmp/cron_test/. My wget code is

```
wget --random-wait -r -p -nd -e robots=off -A".pdf" -U mozilla http://math.stanford.edu/unde
```

So is the

wget

share im

Similar
23:47

add a co

3 Answers

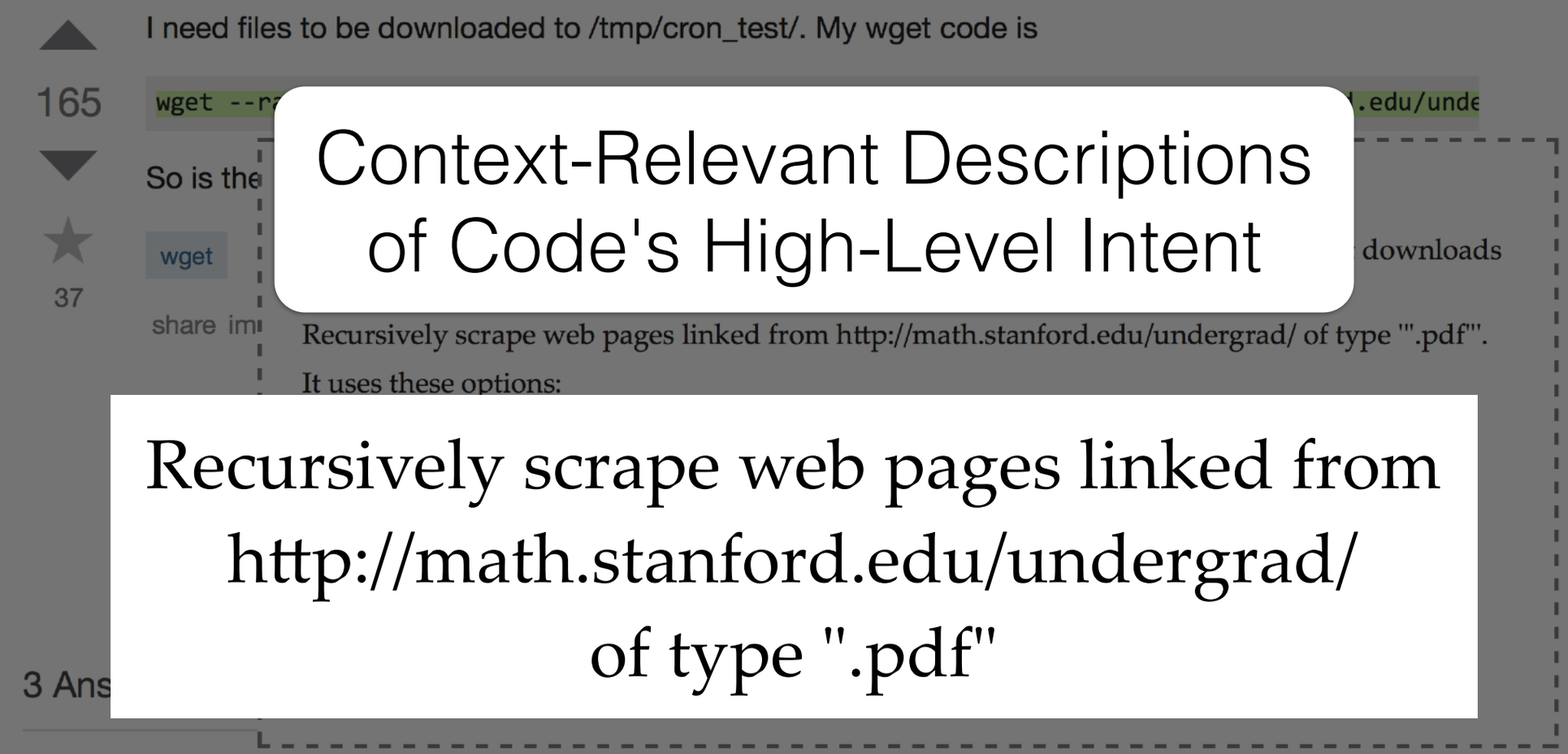
You found a *wget* command.

wget is a Terminal command you run to download a page from the Internet. Here, it downloads content from http://math.stanford.edu/undergrad/.

Recursively scrape web pages linked from http://math.stanford.edu/undergrad/ of type '".pdf"'.

It uses these options:

- `--random-wait`: wait from 0.5*WAIT...1.5*WAIT secs between retrievals.
- `--recursive` (`-r`): specify recursive download.
- `--page-requisites` (`-p`): get all images, etc. needed to display HTML page.
- `--no-directories` (`-nd`): don't create directories.
- `--execute` (`-e`): execute a `.wgetrc'-style command (COMMAND=robots=off).
- `--accept` (`-A`): ".pdf" is a comma-separated list of accepted extensions.
- `--user-agent` (`-U`): identify as mozilla instead of Wget/VERSION.



Context-Relevant Descriptions of Code's High-Level Intent

Recursively scrape web pages linked from http://math.stanford.edu/undergrad/ of type ".pdf"

I need files to be downloaded to /tmp/cron_test/. My wget code is

```
wget --random-wait -r -p -nd -e robots=off -A".pdf" -U mozilla http://math.stanford.edu/unde
```

So

Argument-by-Argument Description of Low-Level Syntax

It uses these options:

- `--recursive` (`-r`): specify recursive download.
- `--user-agent` (`-U`): identify as mozilla instead of Wget/VERSION.

```
1  $(function() {
2      $( ".content_container_7 button" ).button({
```

You found a CSS selector.

The selector '.content_container_7 button' chooses buttons from elements of class 'content_container_7'.

If you haven't seen them before, selectors pick sections of HTML pages by their names or properties. Once you've 'grabbed' elements with a selector, you can manipulate them, like changing their appearance or text.

Here's an example of what this selector will find:

```
<div class="content_container_7">
    <button>
    </button>
</div>
```

n along with

This CSS

specify

button has

Synthesized Prose Explanations

The selector '.content_container_7 button' chooses buttons from elements of class 'content_container_7'.

Synthesized Prose Explanations

```
".content_container_7 button"
```

The selector '.content_container_7 button' chooses buttons from elements of class 'content_container_7'.

Usage Examples

```
<div class="content_container_7">
    <button>
    </button>
</div>
```

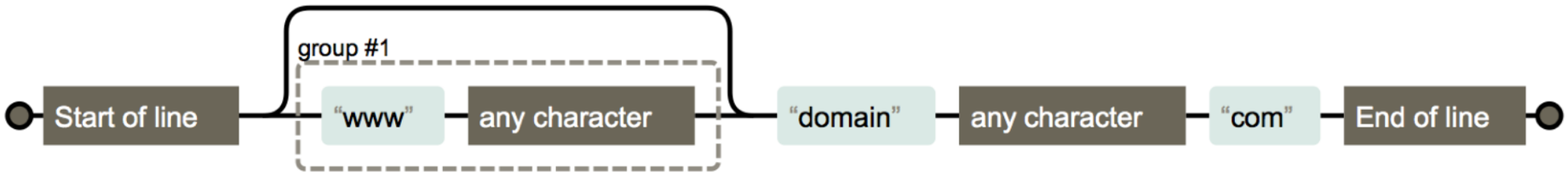
I tried using:

```
RewriteEngine On
RewriteCond %{HTTP_HOST} ^(www.)?domain.com$
RewriteRule ^(/)?$ domain/www [L]
```

You found a regular expression.

Regular expressions are patterns written to match classes of strings. By altering the pattern, you can match different types of strings, like URLs or e-mail addresses.

You can get a feeling for what this expression matches by reading the diagram from left to right:



The pattern ^(www.)?domain.com$ matches strings including:

wwwpdomain=com domain|com

This question came from our site for professional and enthusiast programmers.

I tried using:

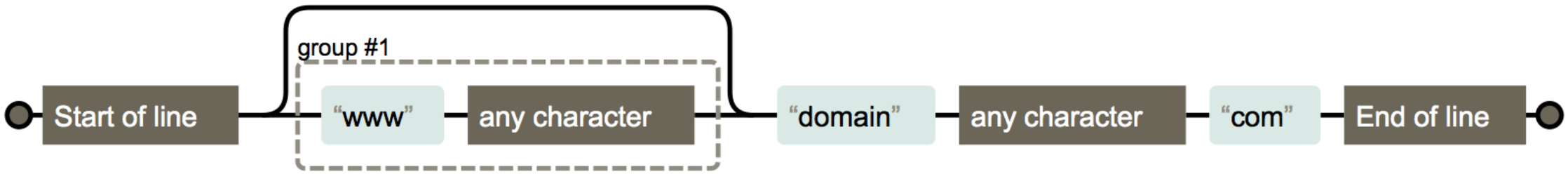
Diagrams

Community-Developed Tools

You found

Regular ex an match different types of str

You can get a feeling for what this expression matches by reading the diagram from left to right:



You can get a feeling for what this expression matches by reading the diagram from left to right:

I tried using:

```
RewriteEngine On
RewriteCond %{HTTP_HOST} ^(www.)?domain.com$
RewriteRule ^(/)?$ domain/www [L]
```

You found

Regular ex an match different types of str

You can get a feeling for what this expression matches by reading the diagram from left to right:

The pattern ^(www.)?domain.com$ matches strings including:

wwwpdomain=com domain|com

This question came from our site for professional and enthusiast programmers.

Usage Examples

An Example of Synthesizing Help

```
div#summ a[href^=/video]
```

this selector chooses links
with URLs starting with "/video" from
a container with ID "summ"

```
div#summ a[href^=/video]
```

1. Generate plaintext description of element

`div`

containers

`a`

links

```
div#summ a[href^=/video]
```

2. Generate modifiers to describe IDs, attributes, etc.

`a`	`[href^=/video]`
links	URLs starting with "/video"
`div`	`#summ`
containers	ID "summ"

```
div#summ a[href^=/video]
```

2. Generate modifiers to describe IDs, attributes, etc.

`a[href^=/video]`

links with URLs starting with "/video"

`div#summ`

a container with ID "summ"

```
div#summ a[href^=/video]
```

3. Append descriptions of parents to children with "from" modifiers

`a[href^=/video]`

links with URLs starting with "/video"

`div#summ`

a container with ID "summ"

```
div#summ a[href^=/video]
```

3. Append descriptions of parents to children with "from" modifiers

```
div#summ a[href^=/video]
```

links with URLs starting with "/video" from a container with ID "summ"

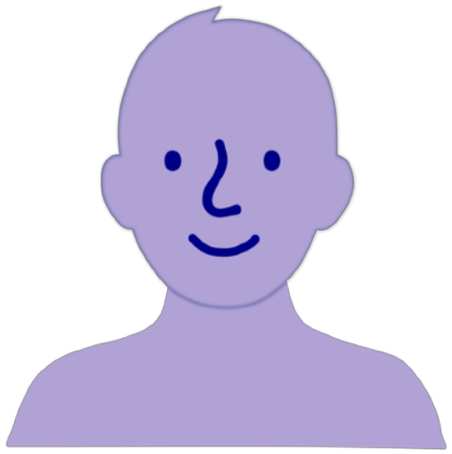
```
div#summ a[href^=/video]
```

3. Append descriptions of parents to children with "from" modifiers

```
div#summ a[href^=/video]
```

this selector chooses
links with URLs starting with "/video" from
a container with ID "summ"

Interacting with Tutorons



Programmer

Navigate

In-Situ Help

Web Browser

Tutorons Library or Addon

Then use the following Wget command to go out and fetch those MP3's:

```
wget -r -l1 -H -t1 -nd -N -np -A.mp3 -erobots=off -i mp3_sites.txt
```

```
form.cmxform fieldset {
  margin-bottom: 10px;
}
form.cmxform legend {
  padding: 0 2px;
  font-weight: bold;
}
form.cmxform label {
  display: inline-block;
  line-height: 1.8;
  vertical-align: top;
}
```

Try this code: $('.my-par').text('');

The CSS selector

Try this code: $('.my-par').text('');

The CSS selector with quotes

Try this code: $('.my-par').text('');

The jQuery selection

Try this code: $('.my-par').text('');

A sloppy selection

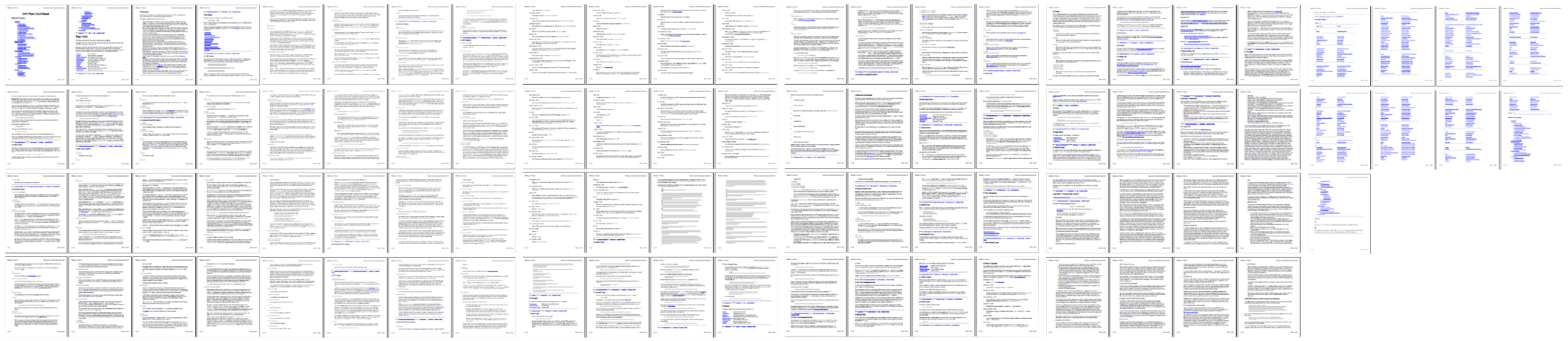
Recursively scrape web pages of type '.mp3' from URLs from the file 'mp3_sites.txt', following links 1 time.

It uses these options:

- `--recursive` (`-r`): specify recursive download.
- `--level` (`-l`): 1 is a maximum recursion depth (inf or 0 for infinite).
- `--span-hosts` (`-H`): go to foreign hosts when recursive.

VS.

```
wget -r -l1 -H -t1 -nd -N -np -A.mp3 -erobots=off -i mp3_sites.txt
```



You found a *wget* command.

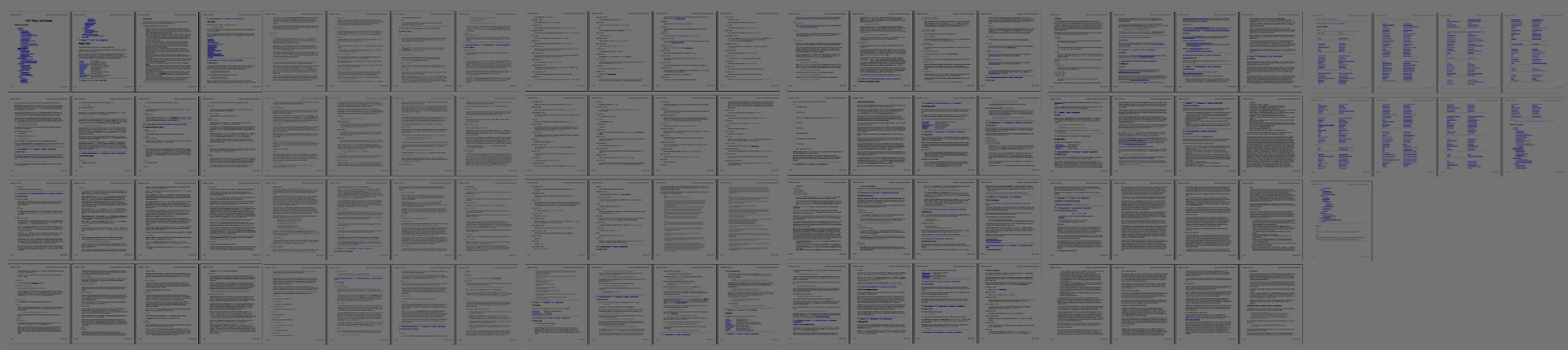
wget is a Terminal command you run to download a page from the Internet. Here, it downloads content from URLs from the file 'mp3_sites.txt'.

Recursively scrape web pages linked from URLs from the file 'mp3_sites.txt' of type '.mp3', following links 1 times.

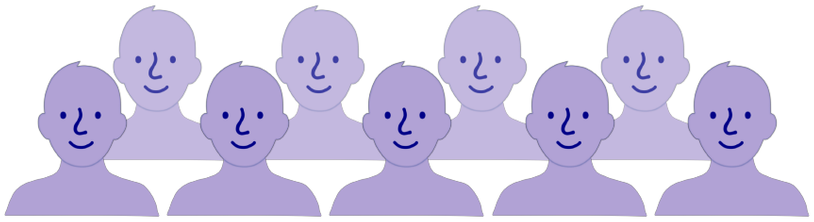
It uses these options:

- `--recursive` (`-r`): specify recursive download.
- `--level` (`-l`): 1 is a maximum recursion depth (inf or 0 for infinite).
- `--span-hosts` (`-H`): go to foreign hosts when recursive.
- `--tries` (`-t`): set number of retries to 1 (0 unlimits).
- `--no-directories` (`-nd`): don't create directories.
- `--timestamping` (`-N`): don't re-retrieve files unless newer than local.

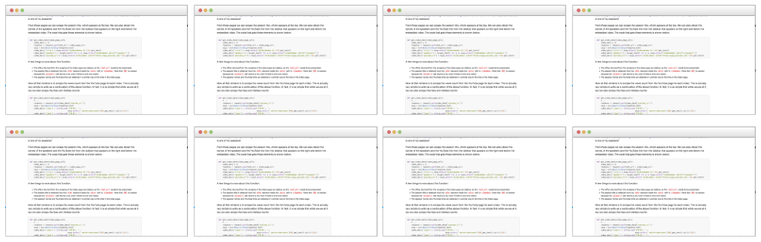
How Useful Are These Explanations?



Experimental Setup



9 programmers



8 wget / CSS modification tasks

You found a *wget* command.

wget is a Terminal command you run to download a page from the Internet. Here, it downloads content from URLs from the file '/tmp/download.txt'.

It uses these options:

- `--continue` (`-c`): resume getting a partially-downloaded file.
- `--background` (`-b`): go to background after startup.
- `--output-file` (`-o`): log messages to /tmp/download.log.
- `--input-file` (`-i`): download URLs found in local or external /tmp/download.txt.
- `--quota` (`-Q`): set retrieval quota to 100m.

vs.

Controlled access to Tutorons

Google

Measured access to external docs

Step 1: Read This Task

Write a CSS selector that selects only elements of class `myInput` .

Step 1: Read This Task

Write a CSS selector that selects only elements of class `myInput`.

Step 2: Look at the Snippet for Clues

Read the snippet below to find code you can modify to do the task. Small fragments of these snippets can be used to do your task. Ask the proctor if you should view accelerators. Afterwards, you can use any other strategies or information you want.

Use:

```
$(".myCheckbox").attr('checked', true); // Deprecated
```

You found a CSS selector.

The selector '.myCheckbox' chooses elements of class 'myCheckbox'.

If you haven't seen them before, selectors pick sections of HTML pages by their names or properties. Once you've 'grabbed' elements with a selector, you can manipulate them, like changing their appearance or text.

Here's an example of what this selector will find:

```
<div class="myCheckbox">
</div>
```

answered Jan 8 '09 at 22:25

bchhun

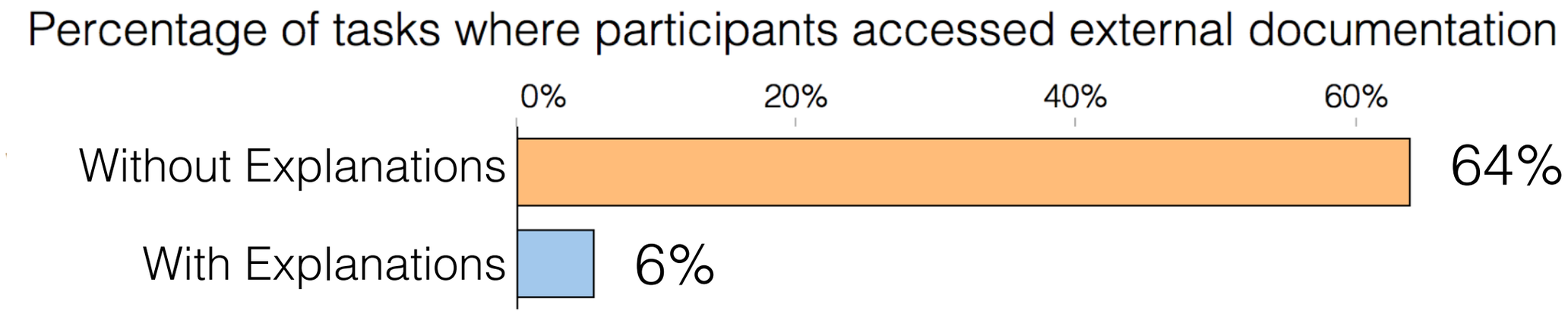
8,252 • 4 • 17 • 22

11 at 23:56

ct all / select none checkbox that needs to check and uncheck all as well as check their state. Instead, I tried @Christopher Harris' answer and that did the trick. â€“ JD Smith Mar 28 '13 at 1:26

show **2** more comments

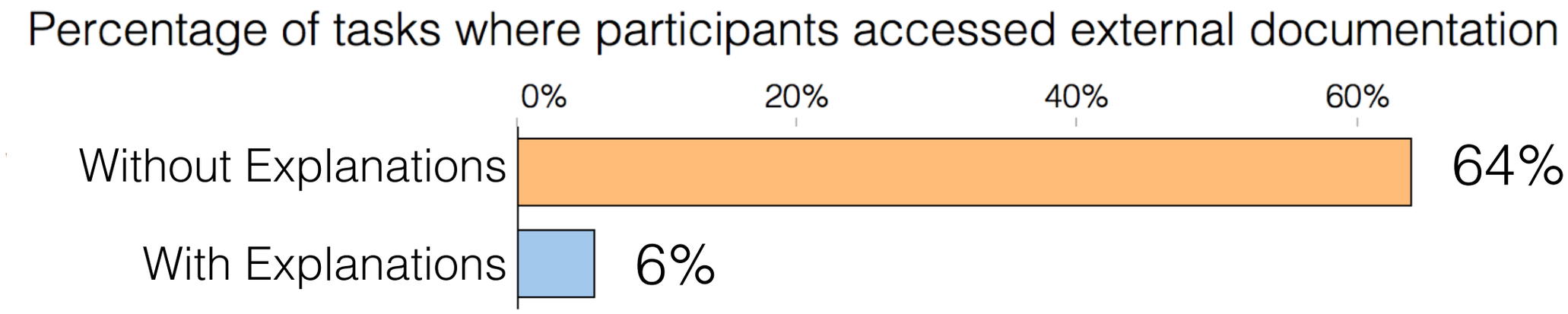
<<<< (You can see the snippet's origin here) >>>>



$p < .0001$ (Fisher's Exact Test, $n = 9$)

Limitations

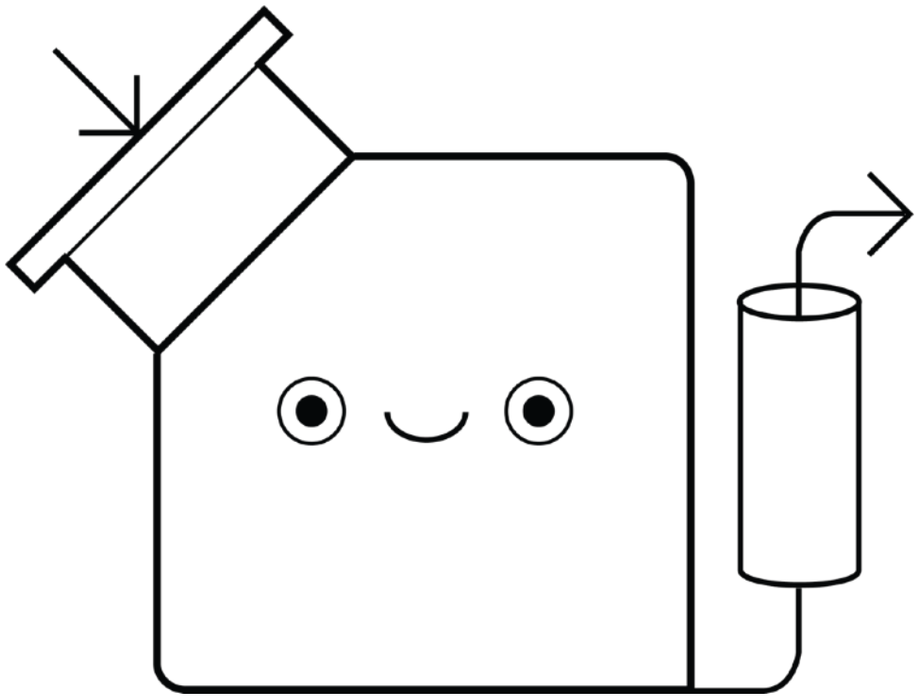
- Small number of participants (n = 9)
- Tasks were designed such that the Tutorons would function properly
- Participants may have been more patient with explanation content than if they were on their own



$p < .0001$ (Fisher's Exact Test, $n = 9$)

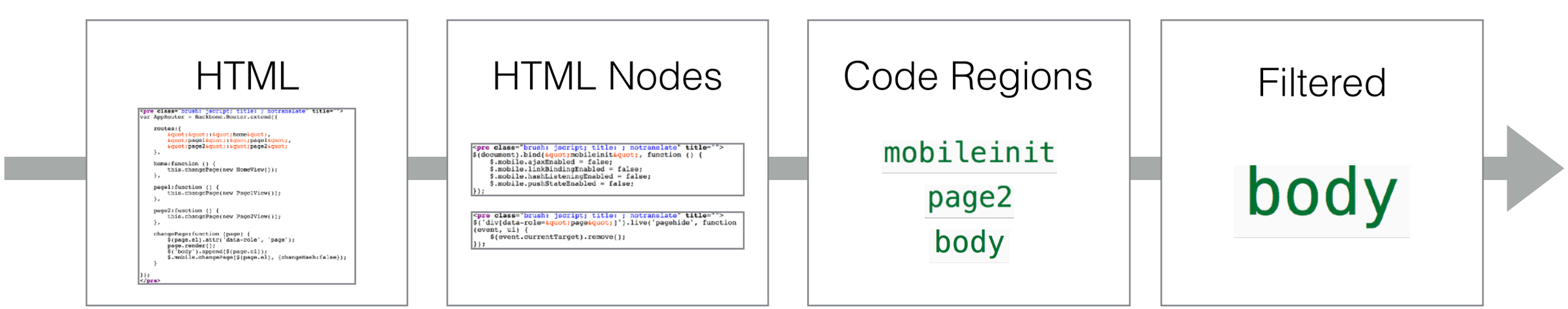
This Talk

- ~~Background~~
- ~~Interacting with Tutorons~~
- **Developing Tutorons**



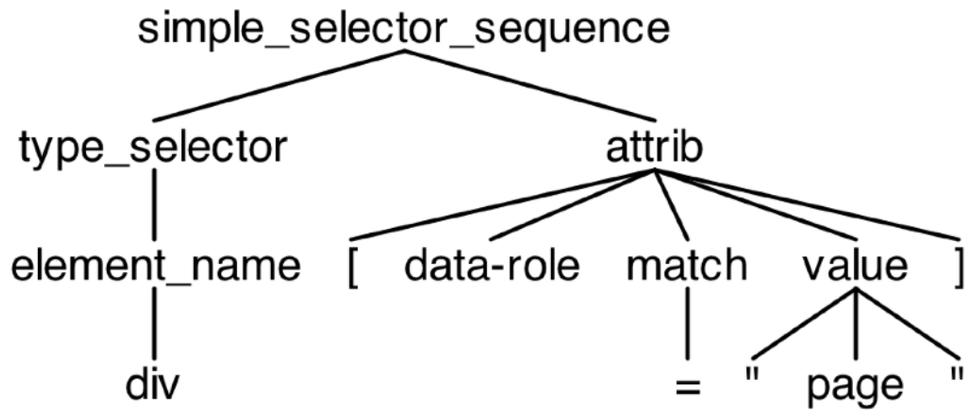

Detect
Parse
Explain

Detecting Code Snippets



Parsing Code Examples

Parse Structures



CSS Selectors

Table of arguments

Short name	Long name	Value
r	recursive	(null)
l	level	2
...	...	...

wget

Generating Explanations and Examples

```
div#summ a[href^=/video]
```

this selector chooses links
with URLs starting with "/video" from
a container with ID "summ"

Counting to Discover Common Usage

```
wget --random-wait -r -p -nd -e robots=off
     A".pdf" -U mozilla
     http://math.stanford.edu/undergrad/
```

Recursively scrape web pages linked from
http://math.stanford.edu/undergrad/
of type ".pdf"

```
wget -r -nH --cut-dirs=2 --no-parent
    --reject="index.html*" http://mysite.com/dir1/dir2/data
```

```
wget -r --no-parent http://mysite.com/configs/.vim/
```

```
wget -r --no-parent --reject "index.html*"
    http://mysite.com/configs/.vim/
```

```
wget -r -nH --cut-dirs=2 --no-parent
    --reject="index.html*" http://mysite.com/dir1/dir2/data
```

Options: `-r`, `-nH`, `--cut-dirs`, `--no-parent`, `--reject`

```
wget -r --no-parent http://mysite.com/configs/.vim/
```

Options: `-r`, `--no-parent`

```
wget -r --no-parent --reject "index.html*"
    http://mysite.com/configs/.vim/
```

Options: `-r`, `--no-parent`, `--reject`

Options: `-r`, `-nH`, `--cut-dirs`, `--no-parent`, `--reject`

`-r`	`--cut-dirs`

`-r`	`--no-parent`

`-r`	`-nH`

...

Options: `-r`, `--no-parent`

`-r`	`--no-parent`

Options: `-r`, `--no-parent`, `--reject`

`-r`	`--no-parent`

`-r`	`--reject`

`--no-parent`	`--reject`

Option 1	Option 2	Count
`-r`	`--no-parent`	3
`-r`	`--reject`	2
`-r`	`-nH`	1
…	…	

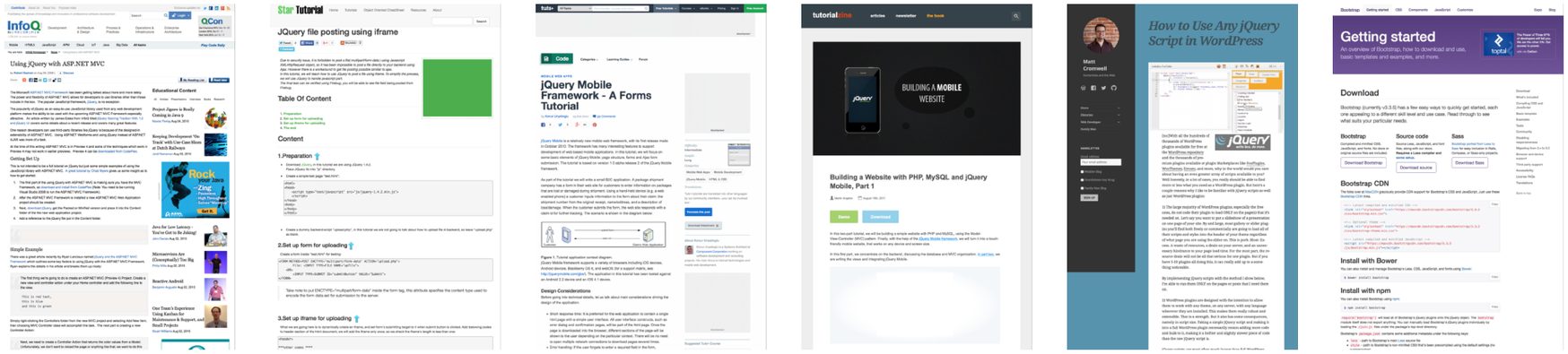
Counting to Discover Common Usage

Option 1	Option 2	Count
-r	-A	28
--user	--password	23
-r	-l	22
...	...	...

Counting to Discover Common Usage

Option 1	Option 2	Count
-r	-A	28
--user	--password	23
-r	-l	22
...	...	...

Testing Detection Quality



50 tutorials / language

```
wget -r -nH --cut-dirs=2 --no-parent --reject="index.html*" http://mysite.com/dir1/dir2/data
wget -r --no-parent http://mysite.com/configs/.vim/
wget -r --no-parent --reject "index.html*" http://mysite.com/configs/.vim/
```

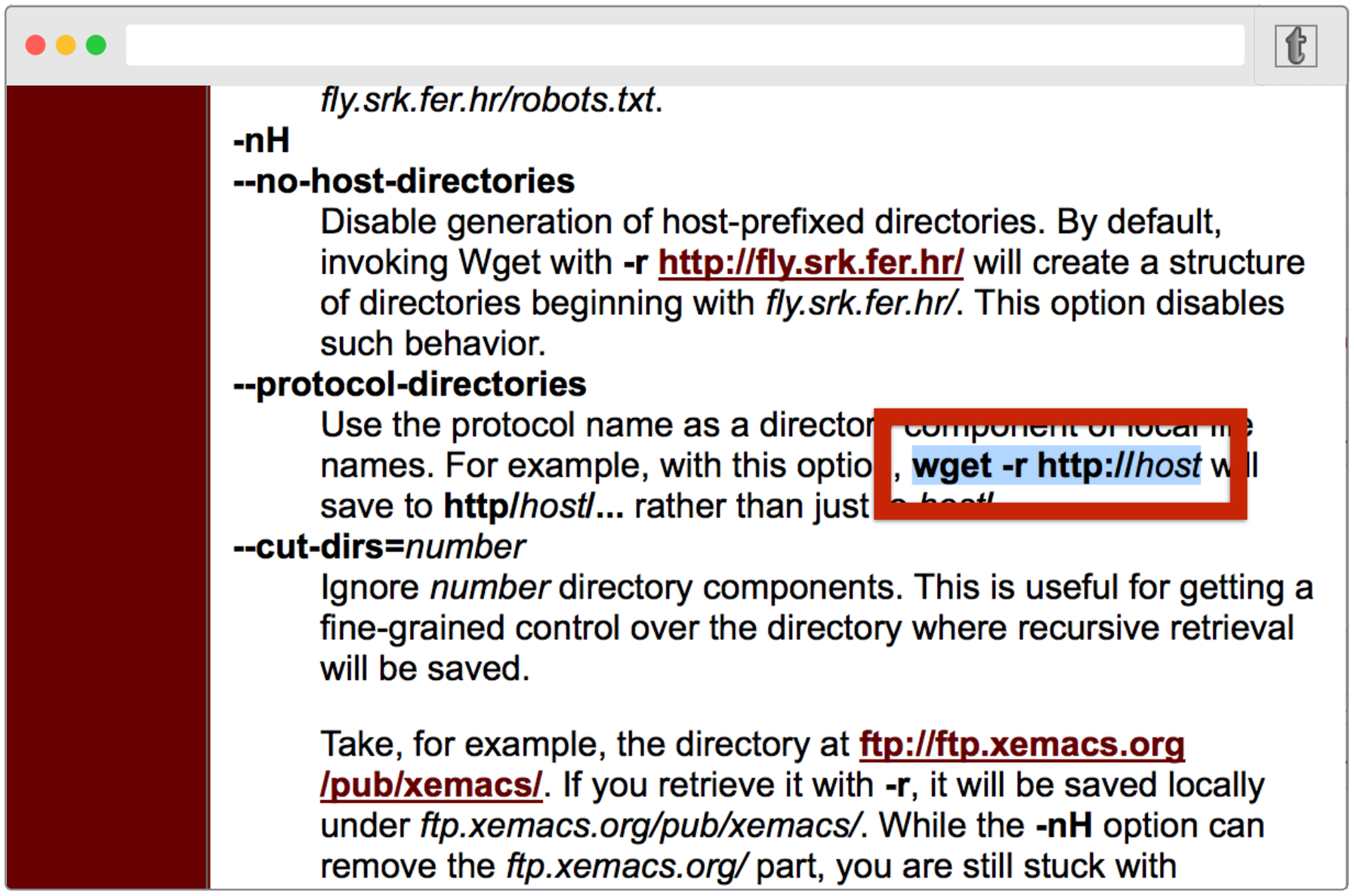
200-500 snippets / language

Detection Accuracy

Language	N	Precision	Recall
wget Unix command	203	**95%**	64%
CSS selectors	466	**80%**	41%
Regular expressions	445	**70%**	14%

Detection Accuracy

Language	N	Precision	Recall
wget Unix command	203	95%	**64%**
CSS selectors	466	80%	**41%**
Regular expressions	445	70%	**14%**



It's Hard to Find Code Within Text

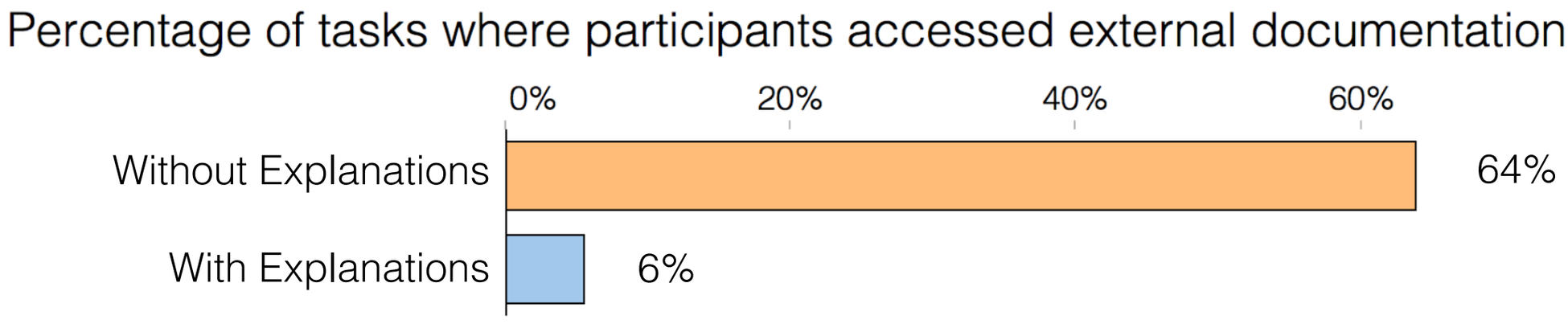
```
set string "0377.255.255.255"
if {[regexp {^(\d+)\.(\d+)\.(\d+)\.(\d+)$}
 && [string is integer $a] && [scan $a %d
 && [string is integer $b] && [scan $b %d
 && [string is integer $c] && [scan $c %d
 && [string is integer $d] && [scan $d %d
```

perl is the cleanest syntax, but if you don't have perl
way to use gawk and components of a regex is to us

```
gawk '/abc[0-9]+xyz/ { print gensub(/.*([0-9]+).
```

output of the sample input file will be

Regular expressions must be found in many languages



$p < .0001$ (Fisher's Exact Test, $n = 9$)

Detection Accuracy

Language	N	Precision	Recall
wget Unix command	203	95%	64%
CSS selectors	466	80%	41%
Regular expressions	445	70%	14%

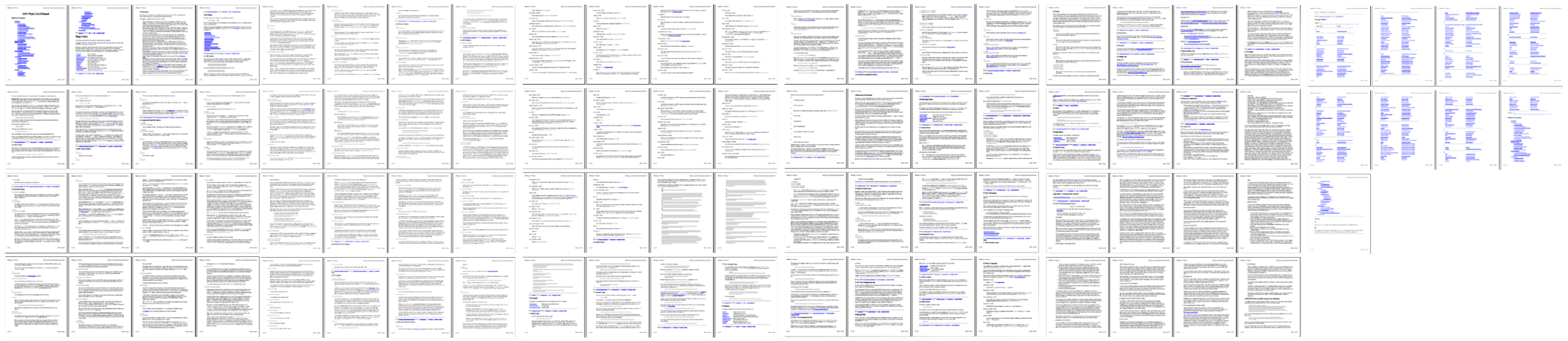
Recursively scrape web pages of type '.mp3' from URLs from the file 'mp3_sites.txt', following links 1 time.

It uses these options:

- `--recursive` (`-r`): specify recursive download.
- `--level` (`-l`): 1 is a maximum recursion depth (inf or 0 for infinite).
- `--span-hosts` (`-H`): go to foreign hosts when recursive.

VS.

```
wget -r -l1 -H -t1 -nd -N -np -A.mp3 -erobots=off -i mp3_sites.txt
```



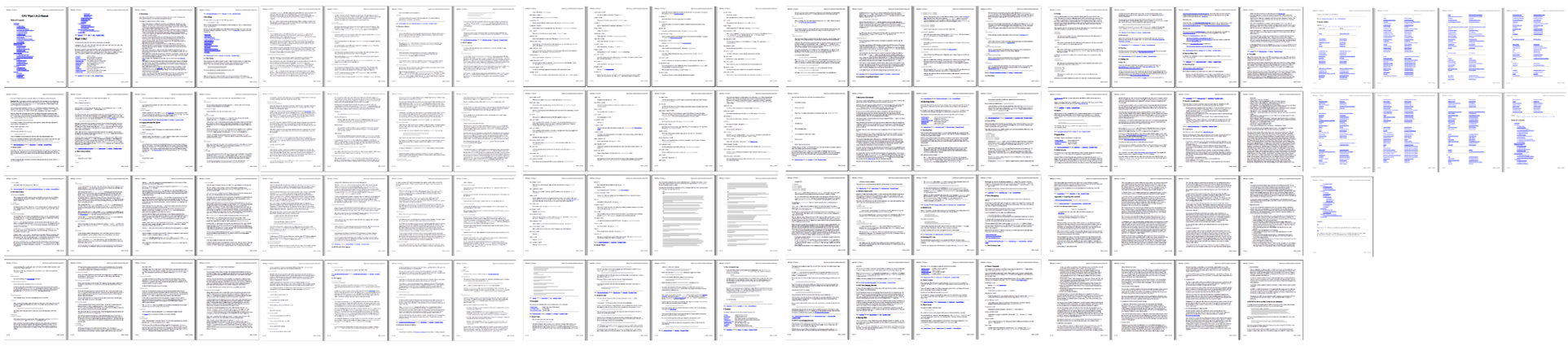
Recursively scrape web pages of type '.mp3' from URLs from the file 'mp3_sites.txt', following links 1 time.

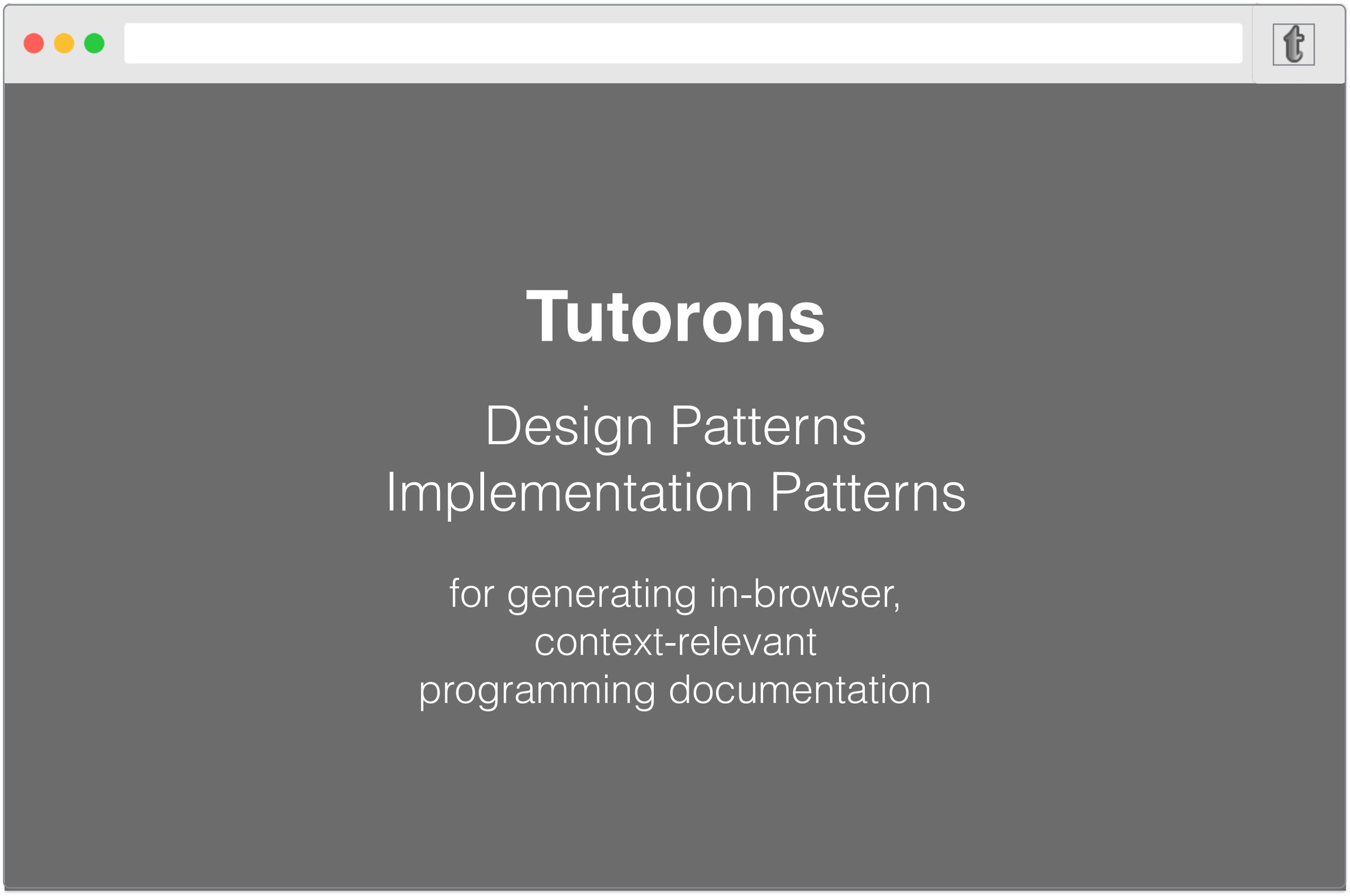
It uses these options:

- `--recursive` (`-r`): specify recursive download.
- `--level` (`-l`): 1 is a maximum recursion depth (inf or 0 for infinite).
- `--span-hosts` (`-H`): go to foreign hosts when recursive.

VS.

```
wget -r -l1 -H -t1 -nd -N -np -A.mp3 -erobots=off -i mp3_sites.txt
```




Tutorons
Design Patterns
Implementation Patterns
for generating in-browser,
context-relevant
programming documentation


www.tutorons.com

Tutorons

```
var conclusion = /Thanks, (VL|HCC) 2015!/g;
```

Andrew Head, Codanda Appachu, Marti A. Hearst, Björn Hartmann
Computer Science Division, UC Berkeley